

# Configuration Management and Product Lines to Enhance the Replication Process in Software Engineering

Edison G. Espinosa

Universidad Politécnica de Madrid Boadilla del Monte 28660

Madrid, Spain

egespinsa1@espe.edu.ec

## ABSTRACT

This research is concerned with the experimental software engineering area, specifically experiment replication. Replication has traditionally been viewed as a complex task in software engineering. This is possibly due to the present immaturity of the experimental paradigm applied to software development. Researchers usually use replication packages to replicate an experiment. However, replication packages are not the solution to all the information management problems that crop up when successive replications of an experiment accumulate.

This research borrows ideas from the software configuration management and software product line paradigms to support the replication process. We believe that configuration management can help to manage and administer information from one replication to another: hypotheses, designs, data analysis, etc. The software product line paradigm can help to organize and manage any changes introduced into the experiment by each replication. We expect the union of the two paradigms in replication to improve the planning, design and execution of further replications and their alignment with existing replications.

Additionally, this research work will contribute a web support environment for archiving information related to different experiment replications. Additionally, it will provide flexible enough information management support for running replications with different numbers and types of changes. Finally, it will afford massive storage of data from different replications. Experimenters working collaboratively on the same experiment must all have access to the different experiments.

## Keywords

Empirical software engineering, experimental replication, replication package, experiment repository.

## 1. INTRODUCTION TO THE RESEARCH

### 1.1 Background

The results of an experiment require consolidation to guarantee their validity and reliability. It is impossible to draw conclusions from a single experiment [11]. Consolidation is achieved by running the experiment several times. This is a way of building

knowledge for use in the software industry [17].

Replication is therefore a key mechanism in the experimental paradigm [7]. Campbell and Stanley [20] explain the role of replication as “The experiments we do today, if successful, will need replication and cross-validation at other times under other conditions before they can become an established part of science, before they can be theoretically interpreted with confidence”. Pragmatically speaking, replication is where other researchers use different samples to try to reproduce research in other contexts [9]. The confidence in the hypotheses under study is raised if the results of the replication are compatible with the results of the original research.

Experimental replication plays a crucial role in experimental software engineering (ESE). However, replication has traditionally been seen as a complex task in ESE. This has been compounded by the difficulty in identifying what information has to be transmitted among experimenters, as well as the complexity of adapting the experiment to a new context [24].

The replication package is a very common mechanism of communication and coordination among experimenters for replicating experiments [1], [2], [24]. A replication package usually includes a set of structured processes, objects, materials, instruments, documents [12], as well as checklists, procedure descriptions, task solutions, etc. [1], [2], [17], [21] concerning the experiment (and sometimes other replications).

Despite the existence of replication packages, there are still very few replications in ESE. Some authors suggest that there is not much motivation for researchers to do replications (they are hard to do and hard to get published). Also, replication packages are, for some reason, not helping replicating experimenters to do their job as well as expected [17]. This suggests that we have not yet hit upon the right solution [10] to the problem of information exchange among experimenters. To complicate things even further, replications tend not to be exact copies of the original experiment, but more or less exact adaptations [7]. An experiment can be replicated with a number of variations depending on what the replication sets out to verify [10]. This makes the information exchange problem even harder, because the replicating experimenters have not only to understand the baseline experiment information but also then to adapt this information to their needs. This diversity implies that even closely related replications do not necessarily have to be identical. This applies to all the associated information, too. Whereas this diversity among related replications exemplifies the idea of the family of experiments [1], it complicates the information and materials exchange problem, as well as the recording of results for later analysis.

The approach that we propose aims to help with the information exchange problem among experimenters and support replicating experimenters. More specifically, it should:

- Help to transfer information so that the replicating experimenters understand the experiment (how to apply the objects, materials, instruments, forms, specifications, etc. of the baseline experiment).
- Be of use for adapting the experiment and configuring changes for the new context.
- Provide an infrastructure that not only stores but also manages (modifies, queries, adapts) the information about the replication process (documents, experimental materials, experimental objects, experimental instruments, forms, specifications, collected data, raw data, analysis results) to offer efficient, reliable and full access.
- Provide information to support decision making for experimental process management and administration.

## 1.2 Motivation

As in the life sciences [16], the experimental processes are complex in ESE. ESE also has the added problem of documentation management [18]. Consequently, the processes for recording, archiving and sharing the descriptions and result of a set of replications of the same experiment are harder. Neither is it easy to obtain and organize the artefacts to support a family of controlled experiments [6]. Specifically, the existence of many replications of an experiment leads to there being different new and obsolete versions of experimental materials. These versions of experimental materials are usually stored at different sites, and have different formats and contents, as they are employed by different experimenters to run different parallel or serial replications.

For decades, other engineering disciplines have proposed processes for transmitting knowledge, avoiding duplicate effort and preventing the same mistakes from being made over again [22]. In order to be able to reproduce an experiment in ESE, it is necessary to transmit information (generally orally and in writing) among experimenters [24]. Experimenters use this information to gather knowledge about the experiment, knowledge of the components of the original experiment (documents, materials, objects, instruments, forms and specifications) that is necessary to be able to modify the experiment either to adapt it to a new context or to check the effect of some variable. There is no universal classification of the information for transmission among experimenters.

Likewise, people have need of decision-making support to run replications [14]. The fact that there is not always efficient and effective access to properly formatted information and versions of materials, etc. is an obstacle to the management and administration that prevents decision making among experimenters in ESE.

We are going to adopt and adapt the software configuration management (SCM) and software product line (SPL) paradigms to enhance the replication process.

We adopt the SCM paradigm by adapting its concepts, relations and instruments for application in the experimental replication process because we think that it will be useful for version management of the information and components (hypotheses,

materials, objects, instruments, forms) used to execute replications.

We believe that the adoption of SPL in the replication process will be useful for the organization and administration of changes made to each experiment replication, enabling new replications to be planned, adapted, designed, released and added to existing replications.

## 2. RELATED WORK

This research is part of the line of technological ESE research concerned with setting up an environment to support experimental and replication processes. We have run an initial review of the literature concerning both the experimental and replication process and support tools in SE. In the future, this initial review will be rounded out by a systematic literature review [13]. The review will cover both SE and other disciplines. However, the work conducted so far has revealed some facts that are described in the following.

To incentivise the experimental and replication processes in SE, some researchers have proposed and developed instruments to transfer information, like publications (sometimes in compliance with guidelines for reporting experiments and/or replications), replication packages, and experiment repositories [24].

In the early replications run in ESE, non-guideline compliant publications on the experiment were only the documents about the original experiment that were transferred [12].

Some researchers, like Lott [15], Basili et al. [2], Solari [19], etc., then proposed static replication packages (repositories including materials) that contain some documents that are necessary for replicating an experiment (materials, objects, instruments, forms, specifications, etc.). Some researchers have added the data collected during the experiment and the materials necessary for training subjects, as well as procedures associated with experiment execution to replication packages. Basili [2] pioneered this work and was later joined by Conradi et al. [5], Shull et al. [17] and Vegas et al. [24].

Some research work in SE provides a technological infrastructure for storing the instruments used to transfer information in different media. Technological infrastructure can range from a simple repository of experimental material to the definition and implementation of tools to support the activities of the experimental process. There have been attempts at storing results (of experiments, case studies and even single experiments) in what might be called experience bases like CEBASE [4], Visek [8], Giants [23] and SIR [6].

We establish a set of criteria. These criteria were obtained from two sources. The first source was experimenters that are members of our research group from whom knowledge on experimentation processes and replication was elicited. The second source was the doctoral student's knowledge of replication gained from experiences supporting experimenters running two replications at both the Escuela Politécnica de Ejército Sede Latacunga (ESPE), Ecuador, and the Universidad Politécnica de Madrid (UPM), Spain, in 2011.

These criteria are useful for identifying the weaknesses of the perceived functionalities with respect to the ideal functionalities elicited from experimenters. Specifically, the proposed evaluation criteria and their associated functionalities are:

- Does it manage experimental materials? This criterion is useful for identifying the functionalities concerning

the storage, recording, querying and traceability of experimental materials.

- Does it manage replication design and execution? This criterion is useful for identifying the functionalities concerning the design, execution and traceability of the replications, replication materials and events occurring during the replication process.
- Does it support experimental administration? This criterion is useful for identifying the functionalities concerning the state of the experiments, replications, events and experimental materials in order to provide decision-making support for experimenters.

Each functionality will be assessed using indicators of completeness and coherence.

- Completeness indicator: Indicator assessing the perceived functionality for an ESE concept. It will be rated on the following scale.
  - None: Value specifying zero functionality.
  - Partial: Value specifying basic functionality (maintenance) concerning storage, retrieval and modification of ESE concepts.
  - Total: Value accounting for all the maintenance functionalities (partial), plus the traceability of the concepts used in ESE.
- Detail level indicator: Indicator used to specify the detail level of the concepts used in ESE.
  - None: Value specifying that the detail level of the ESE concepts is zero.
  - General: Value specifying that the detail level of the ESE concepts is low.
  - Detailed: Value specifying that the detail level of the ESE concepts is high.

## 2.1 Replication Packages

We have identified three main replication packages in the SE field. They were developed by Lott [15], Shull [3] and Solari [19].

Lott's replication package [15] provides the experimental material necessary for replicating an experiment to evaluate software fault detection techniques. It is now active and accessible via web.

Shull's replication package [3] contains information for replicating an experiment on software reading techniques. This package is now active and accessible via web. As well as the material necessary for replicating the experiment, this package

contains additional information on the experiment, such as its design, analysis techniques, materials or experiences from replications run using this package.

Apart from the above experimental packages, we have located a proposal by Solari [19] which provides a set of general- and special-purpose experimental materials and instruments to support replication.

Table 1 shows the concepts, criteria, functionalities and indicators against the values assigned to the analysed packages.

Generally, all three packages have weaknesses with respect to:

- Management of experimental materials, specifically version identification and control.
- Management of replication design and execution.
- Support for experimental administration.

## 2.2 Experimental Information Repositories

We have identified three main repositories in the field of SE: CeBASE [4], ViSEK [8] and SIR [6]:

CeBASE [4] is an experience repository targeting the software industry that applies generic CMMI practices. It was developed by the National Science Foundation (NSF). It does not appear to be in use any longer. This repository is able to organize the results of ESE research in an experience base.

ViSEK [8] is an on-line repository promoting intra- and inter-organizational learning in small- and medium-sized software enterprises in Germany. ViSEK was built, at the initiative of the German government, by a scientific consortium composed of eight research institutes: five Fraunhofer institutes and three research groups based at research universities. This repository is now operational, but has not been upgraded since its creation. It can be used to record experiments, instantiate technologies and exchange experiences through a network interconnecting small and medium-sized software enterprises.

SIR [6] is a repository of Java and C programs and materials for use in experimentation together with testing and analysis techniques. It was built by the Department of Computer Science and Engineering at the University of Nebraska – Lincoln. SIR is now operational and work is ongoing, and there are continual upgrades and add-ons.

Apart from the above repositories, we have located another less ambitious proposal, called Giants [23]. Giants is a proposal for an on-line web portal serving data collection, algorithms, experimental setups, etc.

Table 2 shows the results of applying the criteria and indicators with the values assigned to the analysed repositories.

**Table 1. Evaluation of Replication Packages**

Experimental Package Evaluation Criteria								
Concept	Criterion	Ideal Functionality	Indicators					
			Completeness			Detail Level		
			Scale (total, partial, none)			Scale (detailed, general, none)		
			Lott	Shull	Solari	Lott	Shull	Solari
Experimental materials	Does it manage experimental materials?	Storage of materials used to replicate an experiment (objects, experimental materials, procedures, description/specifications, publications)	partial	partial	partial	detailed	detailed	detailed
		Support for materials versioning (objects, experimental materials, procedures, description/specifications, publications)	none	none	partial	detailed	general	general
		Identification of experimental and other materials to be transmitted for the replication process	none	none	partial	general	general	detailed
Replications	Does it manage replication design and execution?	Details of the designs of executed replications	none	none	partial	none	none	general
		Administration of versions of materials used in an experimental replication	none	none	none	none	none	none
		Administration of versions of experiment replications	none	none	partial	none	none	detailed
		Identification of the states of the executed/executing experimental replications	none	none	none	none	none	none
		Follow-up support for replications	none	none	none	none	none	none
Administration	Does it support experimental administration?	Specification of the state of the materials used in a experiment or replication	none	none	none	none	none	none
		Details of experimental and other materials used in different internal and external experimental replication processes	none	none	none	none	none	general
		Identification of the states of the executed/executing internal and external experimental replications	none	none	none	none	none	general
		Details of events occurring during experiment and replication execution	none	none	partial	none	none	general

**Table 2. Evaluation of Experience Repositories**

Repository Evaluation Criteria										
Concept	Criterion	Ideal Functionality	Indicators							
			Completeness				Detail			
			Scale (total, partial, none)				Scale (detailed, general, none)			
			Visek	CEBASE	GIANTS	SIR	Visek	CEBASE	GIANTS	SIR
Experimental materials	Does it manage experimental materials?	Storage of materials used to replicate an experiment (objects, experimental materials, procedures, description/specifications, publications)	partial	partial	total	total	general	general	general	general
		Support for materials versioning (objects, experimental materials, procedures, description/specifications, publications)	partial	partial	partial	total	general	general	general	general
		Identification of experimental and other materials to be transmitted for the replication process	partial	partial	partial	partial	general	general	general	general
Replications	Does it manage replication design and execution?	Details of the designs of executed replications	partial	none	partial	partial	general	general	general	general
		Administration of versions of materials used in an experimental replication	partial	none	total	partial	general	general	general	general
		Administration of versions of experiment replications	partial	none	total	partial	general	general	general	general
		Identification of the states of the executed/executing experimental replications	partial	none	total	none	general	general	general	general
		Follow-up support for replications	none	none	partial	partial	general	general	general	general
Administration	Does it support experimental administration?	Specification of the state of the materials used in a experiment or replication	partial	none	partial	none	general	general	general	general
		Details of experimental and other materials used in different internal and external experimental replication processes	partial	none	partial	none	general	general	general	general
		Identification of the states of the executed/executing internal and external experimental replications	none	none	none	none	general	general	general	general
		Details of events occurring during experiment and replication execution	none	none	partial	none	general	general	general	general
Aggregation	Is it useful for instantiating experiments?	Definition of experiments	partial	none	partial	total	general	general	general	general
	Is it useful for analyses?	Data analysis	partial	none	partial	partial	general	general	general	general
	Is it useful for determining experiment homogeneity?	Experiment homogeneity	none	none	partial	partial	general	general	general	general
	Does it manage data for aggregation?		none	none	partial	partial	general	general	general	general
		Data organization for aggregation								

Generally, the four repositories have weaknesses with respect to:

- Management of experimental materials, specifically version identification and control.
- Management of replication design and execution.
- Support for experimental administration.

The findings of this review suggest just how limited replication packages and experience repositories are with respect to the applied evaluation criteria. They do not satisfy the ideal functionalities proposed.

### 3. RESEARCH OBJECTIVES

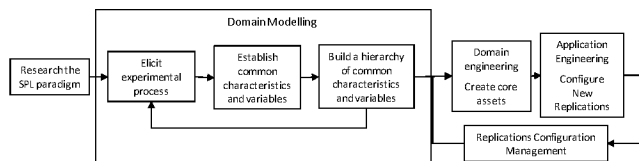
Now that we have analysed the problem and related work, we define the objectives of this research:

- Provide support for the replication process using the components of an experiment, such as materials, objects, instruments, experiment data and analysis.
- Manage the planning, design and execution of the replication.

- Manage the different versions of experimental materials used in the executed replications.
- Implement a distributed, collaborative tool to enable the reuse of experimental elements to support the replication process.

#### 4. RESEARCH APPROACH

We are going to use the Software Product Lines (SPL) paradigm to address the replication process support (adapt the experiment and configure changes for a context) and management problems. We will research the SPL paradigm, analyse a set of documents (elicited from experimenters) on a specific family of experiments and also induce the common characteristics and variables from the different replications run to build a set of core assets (domain engineering). Additionally, we have to establish the processes for configuring new replications (application engineering), the core assets feedback process and, finally, the replication configuration management process. Figure 1 shows the activities for adopting the SPL paradigm.



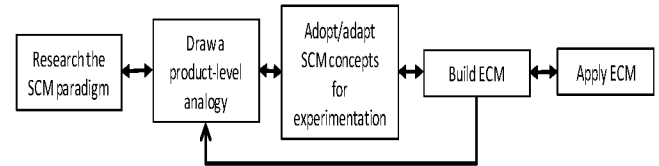
**Figure 1. Diagram of activities for adopting SPL in replication management and support.**

Specifically, the activities to be carried out are:

1. Research the SPL paradigm for application to the process of experiment replication support and management.
2. Elicit the experimental process using interviews and surveys administered to experimenters and analyse the documents on replications of one of a family of experiments executed by the research group.
3. Establish the common materials and variables induced from the replications run by the research group to be able to place a replication within an experiment.
4. Build a hierarchy of the common materials and variables used as input for Activity 3.
5. Create a set of core assets (product line architecture, product line scope, product line configuration plan) for the experimental replication process using the outputs of Activities 3 and 4 as input.
6. Implement the replication configuration process using both core assets and any changes to be made in order to create a replication that is part of a family of experiments.
7. Implement the process for replication process core asset feedback.
8. Implement the configuration management process for the replication version control.

We are going to use the software configuration management (SCM) paradigm to solve the problem of managing previous or successive versions of components of an experiment used in the experimental process. Specifically, we will research the SCM paradigm and develop a product-level analogy between the

software development process and the experimental process. We will adopt and adapt SCM concepts and instruments for application in the SE experimental replication process. We will develop the experimentation configuration management (ECM) and we will apply the ECM solution to a specific group experiment. Figure 2 shows the activities for adopting SCM in the experiment replication process.

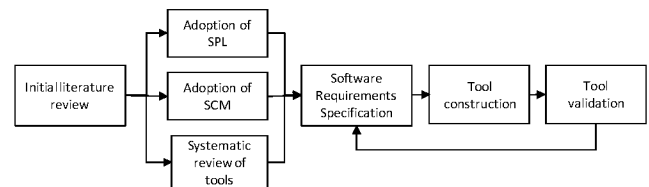


**Figure 2. Diagram of activities for adopting SCM for managing versions of replication materials.**

Specifically, the activities to be carried out are:

1. Investigate the SCM paradigm for application to the material version management process.
2. Draw a product-level analogy between the software development cycle and the experimental research cycle.
3. Adopt and adapt SCM concepts, relations and instruments for the experiment replication process.
4. Build Experimentation Configuration Management (ECM).
5. Apply the solution (ECM) to a research group experiment.

To do this, we will follow the research process detailed in Figure 3.



**Figure 3. General Activities Diagram of Solution Approach.**

This process is composed of the following activities:

1. Initial review of the literature on SPL and SMC.
2. Adoption of SPL in the replication process.
3. Adoption of SCM in the replication process.
4. Systematic review of tools in ESE and other disciplines: repository and experimental packages
5. Draft the software requirements specification (SRS) for the experimental replication process support tool.
6. Construction of the tool to support the experimental replication process with its user manual.
7. Validation of the tool to support the experimental replication process against the SRS. We will also administer questionnaires to both the research group experimenters and peers about tool use experiences.

## 5. SUMMARY OF THE CURRENT STATUS OF THE RESEARCH AND PLANNED NEXT STEPS

So far we have completed the following activities:

- Adoption of the software configuration management (SCM) paradigm. Specifically, we have researched the SCM paradigm. We have drawn a product-level analogy between the software development cycle and the experimental research cycle. We have adopted and adapted SCM concepts, relations, instruments, etc. to experimentation. We have built the experimentation configuration management tool. We have applied the solution to an experiment run by the research group comparing code testing techniques.
- Adoption of the Software Product line (SPL) paradigm. We are now researching the software product line (SPL) paradigm. We have used replications of an experiment executed by our research group for induction, and we have obtained a table and tree hierarchy showing the common materials and variables. Both the table and the tree hierarchy are now being validated by experts.

We plan in the immediate future to carry out several activities in parallel in compliance with the established schedule for 2012. Specifically, the tasks to be executed are:

- Conduct a systematic review of tools.
- Continue the adoption of SPL by developing domain, application and configuration models.
- Apply the SPL proposal to at least one experimental situation.

We will focus in 2013 primarily on developing the experimental replication support tool using incremental prototyping after a technical and operational feasibility study.

- Draft the software requirements specification of the experimental replication process tool.
- Build the experimental replication process support tool.
- Validate the experimental replication process support tool.

Finally, we will put together the final research reports.

## 6. DESCRIPTION OF POINTS ON WHICH WE WOULD LIKE TO GET MOST ADVICE

Any feedback would be welcome, although we are interested in:

- What other information transmission instruments could we research?
- What other research work could be added to the analysis of the related work?
- Do you consider the criteria used to verify the instruments (packages and repositories) for supporting replication to be appropriate?

## 8. REFERENCES

- [1] Basili, V. R., Shull, F., & Filippo, L. (July/ August 1999). Building Knowledge through Families of Experiments. *IEEE Transactions on Software Engineering*, 25(456-473).
- [2] Basili, V., Green, S., Laitenberger, F., Lanubile, F., Shull, F., Sorumgard, S., & Zelkowitz, M. (1996). Packaging Researcher Experience to Assist Replication of Experiments. *ISERN Meeting*, 3-6
- [3] Basili, V., Green, S., Laitenberger, O., Lanubile, F., Shull, F., Sorumgard, S., & Zelkowitz, M. (n.d.). Lab package for the empirical investigation of perspective-based reading. Retrieved from [http://www.cs.umd.edu/projects/SoftEng/ESEG/manual/pbr\\_package/manual.html](http://www.cs.umd.edu/projects/SoftEng/ESEG/manual/pbr_package/manual.html).
- [4] Boehm, B., & Basili, V. (2001). The CeBASE Framework for Strategic Software Development and Evolution. *Proc. of the Third International Workshop on Economics-Driven Software Engineering Research (EDSER-3 2001)*.
- [5] Conradi, R., Basili, V., Carver, J., Shull, F., & G., T. (2001). A Pragmatic Documents Standard for an Experience Library: Roles, Documents, Contents and Structure. technical report, University of Maryland CS-TR-4235.
- [6] Do, H., Elbaum, S., & Rothermel, G. (July 22, 2005). Supporting Controlled Experimentation with Testing Techniques: An Infrastructure and its Potential Impact. *Empirical Software Engineering and International Journal*, 10(4), 405-435.
- [7] Gomez, O. S., Juristo, N., & Vegas, S. (2010). Replications Types in Experimental Disciplines. In *In Proceedings of the International Symposium on Empirical Software Engineering and Measurement (ESEM '10)* (pp. 3:1--3:10). Bolzano-Bozen, Italy.
- [8] Hofmann, B., & Wulf, V. (2003). Building Communities among Software Engineers: The ViSEK Approach to Intra- and Inter-Organizational Learning. *Advances in Learning Software Organizations*, 2640, 25-33.
- [9] Judd, C. M., Smith, E. R., & Kidder, L. H. (1991). *Research Methods in Social Relations*. Orlando, FL, USA.
- [10] Juristo, N., & Gómez, O. (2012). Replication of Software Engineering Experiments. In *Empirical Software Engineering and Verification* (pp. 60-88). Berlin / Heidelberg: Springer.
- [11] Juristo, N. and Vegas, S., "Using Differences among Replications of Software Engineering Experiments to Gain Knowledge," in *In Proceedings of the 2009 3rd*

International Symposium on Empirical Software Engineering and Measurement (ESEM '09), Washington, DC, USA, IEEE Computer Society, 2009, pp. 356-366.

- [12] Kamsties, E., & Lott, C. (1995). An Empirical Evaluation of Three Defect-Detection Techniques. *Proceedings of the Fifth European Software Engineering Conference*, 362-383
- [13] Kitchenham, B., & S., C. (July 2007). Guidelines for performing systematic literature review in software engineering. Technical report, Keele University and University of Durham.
- [14] Lewandowski, A., & Wierzbicki, A. (March 1987). *Theory, Software and Testing Examples for Decision Support Systems*. Laxenburg, Austria.
- [15] Lott, C. M. (n.d.). Comparing reading and testing techniques. Retrieved from <http://web.archive.org/web/20071026204353/http://www.chis-lot.org/work/exp>.
- [16] Paton, N. W. (February 2008). Managing and sharing experimental data: standards, tools and pitfalls. *Biochem Soc Trans*, 36, 33-36.
- [17] Shull, F., Basili, V., Carver, J., Maldonado, J. C., Travassos, G. H., Mendonca, M., & Fabbri, S. (October 2002). Replicating Software Engineering Experiments: Addressing the Tacit Knowledge Problem. *Proceedings of the 2002 International Symposium on Empirical Software Engineering (ISESE'02)*, 7-16.
- [18] Shull, F., Mendonça, M. G., Basili, V., Carver, J., Maldonado, J. C., Fabbri, S., . . . Ferreira, M. C. (2004). Knowledge-Sharing Issues in Experimental Software Engineering. *Empirical Software Engineering*, 9, 111-137.
- [19] Solari, M. (March, 2012). *Propuesta de Paquete de Laboratorio para Experimentos de Ingeniería de Software*. PhD Tesis, Universidad Politécnica de Madrid..
- [20] Stanley, D. T., & Campbell, C. (1973). *Experimental and Quasi-Experimental Designs for Research* Rand McNally.
- [21] Steřik, A., Siebert, S., Melissa, S., & Slattery, K. (2011). An Empirical Comparison of the Accuracy Rates of Novices using the Quorum, Perl, and Randomo Programming Languages. In *Proceedings of the 3rd ACM SIGPLAN workshop on Evaluation and usability of programming languages and tools (PLATEAU '11)* (pp. 3-8). Portland, Oregon, USA: ACM.
- [22] Terveen, L., Selfridge, P., & Long, M. (1995). Living Design Memory' - Framework, Implementation, Lessons Learned. *Human-Computer Interaction*, 10: (1), 1-37..
- [23] Vanschoren, J., & Blockeel, H. (September 2009). Stand on the Shoulders of Giants: Towards a Portal for Collaborative Experimentation in Data Mining. *International Workshop on Third Generation Data Mining at ECML PKDD*, 1:88-89.
- [24] Vegas, S., Juristo, N., Moreno, A., Solari, M., &

Letelier, P. (2006). Analysis of the influence of communication between researchers on experiment replication. In *Proceedings of the International Symposium on Empirical software engineering (ISESE '06)* (pp. 28-37). Rio de Janeiro, Brazil: ACM.